Computation of symbol-wise mutual information in transmission systems with LogAPP decoders and application to EXIT charts

Land, Ingmar; Hoeher, Peter A.; Gligorevic, Snjezana

Published in:
Proceedings of the International ITG Conference on Source and Channel Coding, Erlangen

Publication date:
2004

Document Version
Publisher's PDF, also known as Version of record

Computation of Symbol-Wise Mutual Information in Transmission Systems with LogAPP Decoders and Application to EXIT Charts

Ingmar Land[1], Peter A. Hoeher[1], Snježana Gligorević[2]

[1] Information and Coding Theory Lab, Faculty of Engineering, University of Kiel, Kaiserstr. 2, D-24143 Kiel, Germany. WWW: www.tf.uni-kiel.de/ict. Email: {il,ph}@tf.uni-kiel.de.

[2] Institute for Communications Technology, Darmstadt University of Technology, Merckstr. 25, D-64283 Darmstadt, Germany. WWW: www.nesi.e-technik.tu-darmstadt.de. Email: sng@nesi.tu-darmstadt.de.

Abstract

The symbol-wise mutual information between the binary inputs of a channel encoder and the soft-outputs of a LogAPP decoder, i.e., the a-posteriori log-likelihood ratios (LLRs), is analyzed. This mutual information can be expressed as the expectation of a function of solely the absolute values of the a-posteriori LLRs. This result provides a simple and elegant method for computing the mutual information by simulation. As opposed to the conventional method, explicit measurements of histograms of the soft-outputs are not necessary. In fact, online estimation is possible, and bits having different statistical properties need not be treated separately. As a direct application, the computation of extrinsic information transfer (EXIT) charts is considered.

1 Introduction

In many transmission systems the receiver consists of several processing stages. Due to a well-known result of information theory, the sequences at the outputs of the individual processing stages should not only comprise hard decisions of the transmitted symbols, but also reliability information. A hard decision and its reliability value are usually represented by a single soft value. To ensure that a maximum amount of information is passed, soft-in soft-out processors should be employed in each stage.

The APP (a-posteriori probability) decoder is such a processor, where the soft inputs and the soft outputs are a-posteriori probabilities of the code symbols and/or of the binary information symbols (in the following referred to as info bits). This kind of decoder is optimum in the sense that decisions based on its outputs minimize the bit error rate.

APP decoding may be implemented by means of the BCJR algorithm [1] or in the log-domain by means of the *LogAPP algorithm* [2] (also known as LogMAP), which works directly with log-likelihood ratios (LLRs) and offers practical advantages, such as numerical stability. The outputs of the LogAPP algorithm are a-posteriori LLRs of the code bits and/or of the info bits. In this paper, we will restrict to LogAPP decoding.

The bit error rate can be estimated by using only the absolute values of the a-posteriori LLRs; knowledge of the info bits is *not* necessary [3], [4]. In this paper we will investigate the mutual information between the inputs of the channel encoder (the info bits) and the outputs of the LogAPP decoder (the a-posteriori LLRs). It will be shown that this mutual information can also be computed by using only the absolute values of the LLRs. This is due to the fact that an a-posteriori LLR contains the same information about a transmitted binary symbol as the whole received sequence does, i.e., the a-posteriori LLR is a sufficient statistic of the received sequence.

In [5], the mutual information between blocks of info bits and the respective APPs is investigated. In [6], the information rate between the info bit sequence and the sequence of APPs is addressed. As opposed to [5], [6], we will restrict ourselves to the analysis of the average mutual information between a single info bit and its corresponding a-posteriori LLR, i.e., the *average symbol-wise mutual information* between encoder input and decoder output.

Another closely related topic was investigated in [7]–[10]: The authors applied APP detectors to compute the symmetric information rate of channels with memory. In these papers, the information rate between the *channel input* and the *channel output* is considered. As opposed to this, the present paper deals with the mutual information between the *encoder inputs* and the *decoder outputs*.

A direct application of the method proposed in this paper is a simple and convenient technique to compute extrinsic information transfer (EXIT) charts [11]. These charts describe the input-output behavior of a soft-in soft-out decoder in terms of mutual information, and they can be used to predict the behavior of the iterative decoder for a concatenated code. For further information on the EXIT chart technique, we would

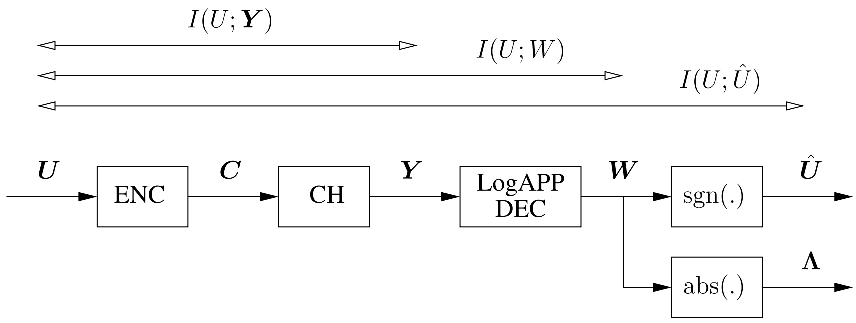

Fig. 1. Transmission system and mutual information under consideration.

like to refer the reader to [11] and related papers.

The conventional method for computing the EXIT chart for a component code of the concatenated code is as follows. Firstly, the histogram of the soft-outputs of the decoder is determined by simulation. Then, the mutual information is computed numerically based on this histogram. If LogAPP decoders (or equivalently APP decoders) are employed as decoders for the component codes, then the computation can be done in only one step by applying the method derived in this paper. The histogram of the soft-outputs need not be determined explicitly.

The paper is structured as follows: In Section 2, the transmission system under consideration is introduced. Furthermore, the type of mutual information addressed in this paper is defined. Section 3 and Section 4 deal with the theoretical background. In the former, some important properties of LLRs are derived, and in the latter, several theorems on mutual information for the case of LogAPP decoders are given and proven. In Section 5, a simple and convenient method for computation of the average symbol-wise mutual information is presented. In Section 6, this method is applied to EXIT chart computation. Finally, conclusions are drawn in Section 7.

2 Transmission System and Notation

Throughout this paper, random variables will be denoted by upper-case letters, and their realizations will be denoted by the corresponding lower-case letters. Vector-valued random variables or realizations will be written boldface.

The transmission system under consideration is depicted in Fig. 1. The info bits $u_k \in \{-1,+1\}$, $k \in \{0,1,\ldots,K-1\}$, are assumed to be independent and uniformly distributed, i.e.,

$$p_U(+1) = p_U(-1) = \frac{1}{2}. \tag{1}$$

They form the info word $\boldsymbol{u} = [u_0, u_1, \ldots, u_{K-1}] \in \{-1,+1\}^K$ of length K. The encoder, the channel, and the decoder are as follows.

Encoder: A linear encoder (ENC) maps the info word $\boldsymbol{u}$ onto the code word $\boldsymbol{c} \in \mathcal{C} \subset \mathbb{Q}^N$ of length N, where $\mathcal{C}$ denotes the code, and $\mathbb{Q}$ denotes the code symbol alphabet.

Channel: The code word is transmitted over a discrete symmetric (not necessarily memoryless) channel (CH), leading to the received word $\boldsymbol{y} \in \mathbb{Y}^N$, where $\mathbb{Y}$ denotes the channel output alphabet.

Decoder: The LogAPP decoder (LogAPP DEC) computes the a-posteriori LLR $w_k \in \mathbb{R}$ for each info bit U_k, $k \in \{0,1,\ldots,K-1\}$. Each a-posteriori LLR is separated into the hard decision, $\hat{u}_k = \operatorname{sgn}(w_k)$, and the reliability of this decision, $\lambda_k = \operatorname{abs}(w_k)$. The words of w_k, $\hat{u}_k$, and λ_k are denoted by $\boldsymbol{w}$, $\hat{\boldsymbol{u}}$, and $\boldsymbol{\lambda}$, respectively.

Note that the transmission system is quite general. The main assumption is that a LogAPP decoder is available.

This paper deals with the mutual information between the encoder input symbols U_k and the decoder output values W_k, $k \in \{0,1,\ldots,K-1\}$. For clarity, we would like to point out the difference between two "types" of mutual information [12]. The average word-wise mutual information (per info symbol) is defined as

$$\bar{I}^{\text{word}} := \frac{1}{K} I(\boldsymbol{U};\boldsymbol{W}), \tag{2}$$

whereas the average symbol-wise mutual information is defined as

$$\bar{I}^{\text{symbol}} := \frac{1}{K} \sum_{k=0}^{K-1} I(U_k;W_k). \tag{3}$$

The second type of mutual information is also used in the EXIT chart method. Due to the well-known information theoretical result that "memory increases capacity", the two types of mutual information are related as $\bar{I}^{\text{word}} \geq \bar{I}^{\text{symbol}}$ [12]. This paper addresses solely the average symbol-wise mutual information.

3 Log-Likelihood Ratios and Properties

In the following two sections, the indices of u_k and w_k will be omitted for convenience. The derivations are valid for all $k \in \{0, 1, \dots, K-1\}$.

The a-priori log-likelihood ratio (LLR) of the info bit U is defined as

$$L(U) := \ln \frac{p_U(+1)}{p_U(-1)}, \tag{4}$$

where $\ln x$ denotes the natural logarithm $\log_e x$. Note that (1) can be written equivalently as $L(U) = 0$. Similarly, the a-posteriori LLR $L(U|\boldsymbol{y})$ and the channel LLR $L(\boldsymbol{y}|U)$ are defined as

$$L(U|\boldsymbol{y}) := \ln \frac{p_{U|\boldsymbol{Y}}(+1|\boldsymbol{y})}{p_{U|\boldsymbol{Y}}(-1|\boldsymbol{y})} \tag{5}$$

and

$$L(\boldsymbol{y}|U) := \ln \frac{p_{\boldsymbol{Y}|U}(\boldsymbol{y}|+1)}{p_{\boldsymbol{Y}|U}(\boldsymbol{y}|-1)}. \tag{6}$$

(We adopt the short-hand notations $L(U|\boldsymbol{y}) := L(U|\boldsymbol{Y} = \boldsymbol{y})$ and $L(\boldsymbol{y}|U) := L(\boldsymbol{Y} = \boldsymbol{y}|U)$ for convenience.) According to the chain rule [13], these LLRs are related as

$$L(U|\boldsymbol{y}) = L(U) + L(\boldsymbol{y}|U)\,. \tag{7}$$

The a-posteriori LLR is regarded as a random variable W with realization

$$w := L(U|\boldsymbol{y})\,. \tag{8}$$

By combining (1), (5), and (8), the a-posteriori probability of U can be written as

$$p_{U|\boldsymbol{Y}}(u|\boldsymbol{y}) = \frac{1}{1 + e^{-wu}}\,, \tag{9}$$

where $u \in \{-1, +1\}$ and $w \in \mathbb{R}$ (cf. [3], [4]).

For computation of the mutual information between U and W, the conditional probabilities of U given W are required. Since the LogAPP decoder provides only the conditional LLR for U given $\boldsymbol{y}$, we have to establish a relation between those two. This is done in the following.

Theorem 1
Let $w = L(U|\boldsymbol{y})$ and $L(U) = 0$. Then,

$$L(U|w) = L(U|\boldsymbol{y})\,.$$

Proof: Considering (7) and $L(U) = 0$, it follows that $w = L(\boldsymbol{y}|U)$. With (6), we get

$$p_{\boldsymbol{Y}|U}(\boldsymbol{y}|+1) = e^{w}\, p_{\boldsymbol{Y}|U}(\boldsymbol{y}|-1)\,.$$

The conditional pdfs of w with respect to u are then related as

$$\begin{aligned} p_{W|U}(w|-1) &= \int_{\boldsymbol{y}:w} p_{\boldsymbol{Y}|U}(\boldsymbol{y}|-1)\; d\boldsymbol{y} \\ &= \int_{\boldsymbol{y}:w} e^{-w}\, p_{\boldsymbol{Y}|U}(\boldsymbol{y}|+1)\; d\boldsymbol{y} \\ &= e^{-w} \int_{\boldsymbol{y}:w} p_{\boldsymbol{Y}|U}(\boldsymbol{y}|+1)\; d\boldsymbol{y} \\ &= e^{-w}\, p_{W|U}(w|+1)\,, \end{aligned}$$

where "$w : \boldsymbol{y}$" is an abbreviation for "$\boldsymbol{y} \in \{\boldsymbol{y}' \in \mathbb{Y}^N : L(U|\boldsymbol{y}') = w\}$". According to (6), we obtain $L(w|U) = w$. This result combined with $L(U) = 0$ and (7) gives the proof. $\square$

Theorem 1 can easily be transformed into the probability domain by using the definition of the a-posteriori LLR according to (5). Taking (9) into account, we immediately obtain the following equalities.

Corollary 2
Let $w = L(U|\boldsymbol{y})$ and $L(U) = 0$. Then,

$$p_{U|\boldsymbol{Y}}(u|\boldsymbol{y}) = p_{U|W}(u|w) = \frac{1}{1 + e^{-wu}} \tag{10}$$

for $u \in \{-1, +1\}$.

These identities will be utilized in the next section.

4 Mutual Information and LogAPP Decoding

In this section, the relations between the (unconditional) mutual information between U and W, the (unconditional) mutual information between U and $\boldsymbol{Y}$, and the corresponding conditional mutual informations for given Λ are discussed. As in the previous section, the indices of U_k, W_k, and Λ_k will be omitted for convenience.

First, the effect of conditioning the mutual information on the reliability Λ is investigated.

Theorem 3
The mutual information between an info bit U and the received word $\boldsymbol{Y}$ and the mutual information between an info U bit and the respective a-posteriori LLR W do not change when they are conditioned on the reliability Λ, i.e.,

$$\begin{aligned} I(U;\boldsymbol{Y}) &= I(U;\boldsymbol{Y}|\Lambda), \\ I(U;W) &= I(U;W|\Lambda). \end{aligned}$$

Proof: First equality: With λ being a function of $\boldsymbol{y}$ and with the chain rule of entropy, we obtain

$$\begin{aligned} H(\boldsymbol{Y}) &= H(\boldsymbol{Y},\Lambda) \\ &= H(\boldsymbol{Y}|\Lambda) + H(\Lambda) \end{aligned}$$

and

$$\begin{aligned} H(\boldsymbol{Y}|\boldsymbol{U}) &= H(\boldsymbol{Y},\Lambda|U) \\ &= H(\boldsymbol{Y}|U,\Lambda) + H(\Lambda|U). \end{aligned}$$

Taking into account that $H(\Lambda|U) = H(\Lambda)$, the mutual information can then be written as

$$\begin{aligned} I(U;\boldsymbol{Y}) &= H(\boldsymbol{Y}) - H(\boldsymbol{Y}|\boldsymbol{U}) \\ &= H(\boldsymbol{Y}|\Lambda) - H(\boldsymbol{Y}|U,\Lambda) \\ &= I(U;\boldsymbol{Y}|\Lambda)\,. \end{aligned}$$

Second equality: Considering that λ is a function of $\boldsymbol{y}$, the second part of the theorem can be proven similarly. $\square$

For a given realization of Λ, we get the following relation.

Theorem 4
For every $\lambda = |L(U|\boldsymbol{y})|$, the following equality holds:

$$I(U;\boldsymbol{Y}|\lambda) = I(U;W|\lambda)\,.$$

Proof: Since λ is a function of $\boldsymbol{y}$, we have $p_{U|\boldsymbol{Y}\Lambda}(u|\boldsymbol{y},\lambda) = p_{U|\boldsymbol{Y}}(u|\boldsymbol{y})$, and since λ is a function of w, we have $p_{U|W\Lambda}(u|w,\lambda) = p_{U|W}(u|w)$. These equalities and Corollary 2 give

$$\begin{aligned} p_{U|\boldsymbol{Y}\Lambda}(u|\boldsymbol{y},\lambda) &= p_{U|\boldsymbol{Y}}(u|\boldsymbol{y}) = \\ &= p_{U|W}(u|w) = p_{U|W\Lambda}(u|w,\lambda). \end{aligned}$$

Thus we have

$$\begin{aligned} I(U;\boldsymbol{y}|\lambda) &= \sum_{u=\pm 1} p_{U|\boldsymbol{Y}\Lambda}(u|\boldsymbol{y},\lambda)\,\mathrm{ld}\,\frac{p_{U|\boldsymbol{Y}\Lambda}(u|\boldsymbol{y},\lambda)}{p_{U|\Lambda}(u|\lambda)} \\ &= \sum_{u=\pm 1} p_{U|W\Lambda}(u|w,\lambda)\,\mathrm{ld}\,\frac{p_{U|W\Lambda}(u|w,\lambda)}{p_{U|\Lambda}(u|\lambda)} \\ &= I(U;w|\lambda), \end{aligned} \tag{11}$$

where $\mathrm{ld}\,x$ denotes the binary logarithm $\log_2 x$. Taking the conditional expectation over $\boldsymbol{y}$ and w with respect to λ, we get the following chain of equalities:

$$\begin{aligned} I(U;\boldsymbol{Y}|\lambda) &= \int_{\boldsymbol{y}} p_{\boldsymbol{Y}|\Lambda}(\boldsymbol{y}|\lambda)\; I(U;\boldsymbol{y}|\lambda)\; d\boldsymbol{y} \\ &\overset{(a)}{=} \int_{\boldsymbol{y}:\lambda} p_{\boldsymbol{Y}|\Lambda}(\boldsymbol{y}|\lambda)\; I(U;\boldsymbol{y}|\lambda)\; d\boldsymbol{y} \\ &\overset{(b)}{=} \int_{\boldsymbol{y}:\lambda} p_{\boldsymbol{Y}|\Lambda}(\boldsymbol{y}|\lambda)\; I(U;w|\lambda)\; d\boldsymbol{y} \\ &\overset{(c)}{=} \int_{\boldsymbol{y}:w=\lambda} p_{\boldsymbol{Y}|\Lambda}(\boldsymbol{y}|\lambda)\; I(U;w|\lambda)\; d\boldsymbol{y} + \\ &\quad + \int_{\boldsymbol{y}:w=-\lambda} p_{\boldsymbol{Y}|\Lambda}(\boldsymbol{y}|\lambda)\; I(U;w|\lambda)\; d\boldsymbol{y} \\ &= I(U;W=+\lambda|\lambda)\cdot \\ &\qquad \cdot \int_{\boldsymbol{y}:w=\lambda} p_{\boldsymbol{Y}|\Lambda}(\boldsymbol{y}|\lambda)\; d\boldsymbol{y} + \\ &\quad + I(U;W=-\lambda|\lambda)\cdot \\ &\qquad \cdot \int_{\boldsymbol{y}:w=-\lambda} p_{\boldsymbol{Y}|\Lambda}(\boldsymbol{y}|\lambda)\; d\boldsymbol{y} \\ &\overset{(d)}{=} \sum_{w=\pm\lambda} p_{W|\Lambda}(w|\lambda)\; I(U;w|\lambda) \\ &= I(U;W|\lambda)\,. \end{aligned} \tag{12}$$

The applied relations are: (a) $p_{\boldsymbol{Y}|\Lambda}(\boldsymbol{y}|\lambda) = 0$ for all $\boldsymbol{y}$ which do not lead to λ; (b) Equ. (11); (c) for a given λ, we have either $w = \lambda$ or $w = -\lambda$; (d) $\int_{\boldsymbol{y}:w=\lambda} p_{\boldsymbol{Y}|\Lambda}(\boldsymbol{y}|\lambda) = p_{W|\Lambda}(\lambda|\lambda)$ and $\int_{\boldsymbol{y}:w=-\lambda} p_{\boldsymbol{Y}|\Lambda}(\boldsymbol{y}|\lambda) = p_{W|\Lambda}(-\lambda|\lambda)$. $\square$

From Theorem 4, it follows that $\mathrm{E}_\Lambda\{I(U;\boldsymbol{Y}|\lambda)\} = \mathrm{E}_\Lambda\{I(U;W|\lambda)\}$, which can be written as $I(U;\boldsymbol{Y}|\Lambda) = I(U;W|\Lambda)$. ($\mathrm{E}\{.\}$ denotes expectation.) After applying Theorem 3 to this equality, we get immediately the following result.

Corollary 5
The a-posteriori LLR W of an info bit U contains the same information about this info bit as the received vector $\boldsymbol{Y}$ does, i.e.,

$$I(U;\boldsymbol{Y}) = I(U;W).$$

Note that this is another formulation of the well-known fact that the a-posteriori LLR w is a sufficient statistic of the received word $\boldsymbol{y}$ with respect to the info bit U.

Consider now how the conditional mutual information $I(U;W|\lambda)$ can be expressed by using only the absolute values of the a-posteriori LLRs. For doing so, let first define the following function.

Definition 1
For $x \geq 0$,

$$f_I(x) := \frac{1}{1+e^{+x}}\,\mathrm{ld}\,\frac{2}{1+e^{+x}} + \frac{1}{1+e^{-x}}\,\mathrm{ld}\,\frac{2}{1+e^{-x}}.$$

The meaning of this function is made clear in following theorem.

Theorem 6
Let $w = L(U|\boldsymbol{y})$ and $\lambda = |w|$. Then,

$$I(U;W|\lambda) = f_I(\lambda).$$

Proof: Consider the following chain of equalities:

$$\begin{aligned} I(U;W|\lambda) &= \sum_{w=\pm\lambda} p_{W|\Lambda}(w|\lambda)\; I(U;w|\lambda) \\ &= \sum_{w=\pm\lambda} p_{W|\Lambda}(w|\lambda)\cdot \\ &\qquad \cdot \sum_{u=\pm 1} p_{U|W\Lambda}(u|w,\lambda)\cdot \\ &\qquad\quad \cdot\,\mathrm{ld}\,\frac{p_{U|W\Lambda}(u|w,\lambda)}{p_{U|\Lambda}(u|\lambda)} \\ &\overset{(a)}{=} \sum_{w=\pm\lambda}\sum_{u=\pm 1} p_{W|\Lambda}(w|\lambda)\cdot \\ &\qquad \cdot \frac{1}{1+e^{-wu}}\,\mathrm{ld}\,\frac{2}{1+e^{-wu}} \\ &\overset{(b)}{=} \frac{1}{1+e^{+\lambda}}\,\mathrm{ld}\,\frac{2}{1+e^{+\lambda}} + \\ &\quad + \frac{1}{1+e^{-\lambda}}\,\mathrm{ld}\,\frac{2}{1+e^{-\lambda}}. \end{aligned}$$

The applied relations are: (a) $p_{U|\Lambda}(u|\lambda) = p_U(u) = \frac{1}{2}$ and Corollary 2; (b) $p_{W|\Lambda}(\pm\lambda|\lambda) = 1/2$ due to symmetry. $\square$

This theorem verifies that $I(U;W|\lambda)$ is only a function of the absolute values of the LLRs, as already mentioned in the introduction. Therefore, it depends neither on the actual values of the info bits nor on the channel model. Fig. 2 shows the plot of $I(U;W|\lambda)$ versus λ.

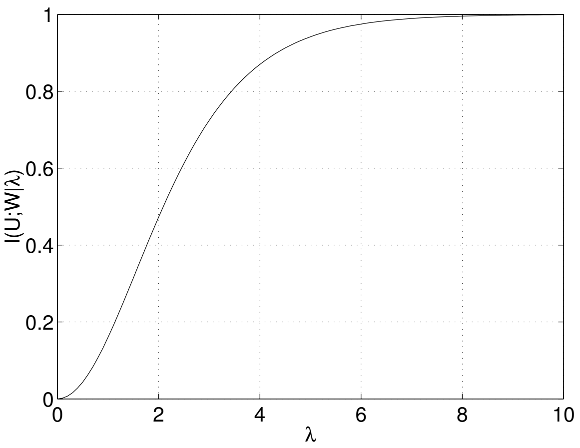

Fig. 2. The conditional mutual information between an info bit U and the corresponding a-posteriori LLR W, $I(U;W|\lambda) = f_I(\lambda)$, versus the absolute value of the a-posteriori LLR, $\lambda = |L(U|\boldsymbol{y})| = |w|$.

For the sake of completeness, we also give the mutual information between the info bit U and its estimate $\hat{U}$:

$$I(U;\hat{U}) = -p_e \operatorname{ld} p_e - (1-p_e) \operatorname{ld} (1-p_e)\,,$$

where $p_e := P(U \neq \hat{U})$ denotes the bit error probability. Using Corollary 2, it is easy to show that

$$\begin{aligned} p_e &= \int_{w<0} \frac{p_{U|W}(+1|w)}{\sum_{u=\pm1} p_{U|W}(u|w)}\, dw \\ &= \int_{w<0} \frac{e^{+w/2}}{e^{+w/2}+e^{-w/2}}\, dw\,. \end{aligned}$$

5 Computation of Mutual Information

The theoretical results found in the previous two sections will now be exploited for computing the mutual information by simulation.

As already stated in Section 2, the average symbol-wise mutual information is defined as

$$\bar{I}^{\text{symbol}} := \frac{1}{K} \sum_{k=0}^{K-1} I(U_k;W_k), \tag{13}$$

where the symbol-wise mutual information for position k is given by

$$\begin{aligned} I(U_k;W_k) &= \mathrm{E}\Big\{ \operatorname{ld} \frac{p_{U_k|W_k}(u|w)}{p_{U_k}(u)} \Big\} = \\ &= \int p_{U_k,W_k}(u,w) \operatorname{ld} \frac{p_{U_k|W_k}(u|w)}{p_{U_k}(u)} dw, \end{aligned} \tag{14}$$

$k \in \{0,1,\ldots,K-1\}$. Since the info bits U are assumed to be independent and uniformly distributed, the value of this mutual information depends only on the pdfs $p_{W_k|U_k}(w|u)$.

For distinquishing between actual values and estimated values, we adopt the following notation: Let $\hat{p}_{W_k|U_k}(w|u)$ denote histograms representing estimates of the actual pdfs $p_{W_k|U_k}(w|u)$. Let further $\hat{I}(U_k;W_k)$ denote estimates of $I(U_k;W_k)$, and let $\hat{\bar{I}}^{\text{symbol}}$ denote an estimate of $\bar{I}^{\text{symbol}}$.

Generally, the pdfs in (14) may depend on k, and thus also $I(U_k;W_k)$ may depend on k. Accordingly, an unbiased simulative estimation of the average symbol-wise mutual information comprises the following three steps:

1) Determine simulatively the histograms $\hat{p}_{W_k|U_k}(w|u)$ for each $k \in \{0,1,\ldots,K-1\}$.
2) Compute numerically $\hat{I}(U_k;W_k)$ based on $\hat{p}_{W_k|U_k}(w|u)$ for each $k \in \{0,1,\ldots,K-1\}$, according to (14).
3) Compute $\hat{\bar{I}}^{\text{symbol}}$ by averaging over $I(U_k;W_k)$ with respect to k, according to (13).

This method is referred to as *three-step method* in the sequel. Note that only positions k having different statistical properties need to be treated separately.

If the statistical properties do not depend on position k, the following *two-step method* is sufficient:

1) Determine simulatively the histogram $\hat{p}_{W|U}(w|u)$.
2) Compute numerically $\hat{I}(U;W)$ based on $\hat{p}_{W|U}(w|u)$, according to (14). Due to the independence of k, we have $\hat{\bar{I}}^{\text{symbol}} = \hat{I}(U;W)$, and averaging according to (13) is not necessary.

Note that the three-step method and the two-step method are not equivalent, if the statistical properties depend on position k. Step 1 and and Step 3 of the three-step method are not interchangable, because Step 2 is a non-linear operation.

The applicability of the two methods shall be discussed for several examples of transmission systems. Binary channel codes and discrete memoryless channels are assumed if not stated otherwise.

- For *tail-biting convolutional codes*, the trellis is not time-varying, and thus the statistical properties are the same for each position k in the trellis. Therefore, the two-step method can be applied to get an unbiased estimate.
- Consider now *terminated convolutional codes* with well-defined initial and final state of the encoder. The trellis is time-varying only at the beginning and at the end, and it is time-invariant in between. Therefore the statistical properties can be assumed to be the same only for positions k in the centre part of the trellis. Correspondingly, only the three-step method will give an unbiased estimate of the mutual information.

  Nevertheless, an estimate based on the two-step method will tend to the true value if the trellis length approaches infinity. Thus, the resulting estimate can still be used as an approximation in cases where the major part of the trellis is time-invariant.

- For *codes having time-varying trellises*, like punctured convolutional codes, time-varying convolutional codes, or block codes (most of which have time-varying trellises), the two-step method can never give an unbiased estimate of the mutual information. Note that this is also the case for higher order modulation schemes. On the other hand, applying the three-step method can become quite complicated, even if positions k having the same statistical properties are treated in groups.

The theorems from the previous section provide a means for computing the average symbol-wise mutual information without explicit determination of histograms and without the necessity of distinguishing between positions k having different statistically properties. The average symbol-wise mutual information can be written as

$$
\begin{aligned}
\bar{I}^{\text{symbol}} &\overset{(a)}{=} \frac{1}{K}\sum_{k=0}^{K-1} I(U_k;W_k) \\
&\overset{(b)}{=} \frac{1}{K}\sum_{k=0}^{K-1} I(U_k;W_k|\Lambda_k) \\
&= \frac{1}{K}\sum_{k=0}^{K-1} \mathrm{E}\big\{I(U_k;W_k|\lambda_k)\big\} \\
&\overset{(c)}{=} \frac{1}{K}\sum_{k=0}^{K-1} \mathrm{E}\big\{f_I(\lambda_k)\big\} \\
&= \mathrm{E}\Big\{\frac{1}{K}\sum_{k=0}^{K-1} f_I(\lambda_k)\Big\} \\
&\overset{(d)}{=} \mathrm{E}\{f_I(\lambda)\}.
\end{aligned}
$$

The applied relations are: (a) Equ. (13); (b) Theorem 3; (c) Theorem 6; (d) since the function $f_I(.)$ does not depend on k, the different positions k need not be treated separately. Thus, we have the following result.

Theorem 7
Let $\lambda = |L(U|\boldsymbol{y})|$ and let function $f_I(.)$ be defined according to Definition 1. The average symbol-wise mutual information between encoder input and LogAPP decoder output is given as

$$\bar{I}^{symbol} = \mathrm{E}\{f_I(\lambda)\}.$$

The evaluation of the mutual information by means of Theorem 7 can be interpreted in the following way: The absolute value of the a-posteriori LLR, Λ, is used to partition the channel between U and W (see Fig. 1) into binary symmetric sub-channels, of which the outputs are either $-\lambda$ or $+\lambda$; let these sub-channels be labeled with λ. For each sub-channel λ, the mutual information is given by $I(U;W|\lambda) = f_I(\lambda)$, which is only a function of λ. The mutual information of the overall channel, $I(U;W)$, is then the expectation of the mutual information of the sub-channels.

Theorem 7 can easily be translated into the following *one-step method* for determining the mutual information by simulation. The only requirement for applicability is that a-posteriori LLRs of the info bits, $L(U|\boldsymbol{y})$, are available. Note that funtion $f_I(x)$ is given in Definition 1.

For each a-posteriori LLR, determine its absolute value $\lambda = |L(U|\boldsymbol{y})|$, and average over the function values $f_I(\lambda)$.

This method has several advantages compared to the three-step method and the two-step method.

1) No histograms need to be determined. This method can rather operate “on-line”, because as soon as a new a-posteriori LLR is available, it can be used to update the current estimate of $\bar{I}^{\text{symbol}}$.
2) The results are exact, i.e., unbiased, even for time-varying trellises.
3) The reliability of the estimate for $\bar{I}^{\text{symbol}}$ can easily be determined. Since the estimate for $\bar{I}^{\text{symbol}}$ is simply the mean of the samples $f_I(\lambda_i)$, its variance is equal to the variance of $f_I(\lambda_i)$ divided by the number of such samples; the variance of $f_I(\lambda_i)$ can easily be computed during simulation. For methods based on histograms, estimation of the reliability is less obvious.

Thus, the one-step method is simple and convenient for all cases, and it is efficient for the cases where the statistical properties of the info bits depend on their positions. The method relies only on the fact that the decoder delivers a-posteriori LLRs.

6 Application to EXIT Charts

In many receivers several soft-in soft-out APP modules detect or decode in an iterative fashion by exchanging extrinsic LLRs. An efficient method to determine the minimum signal-to-noise ratio, at which iterative detection or decoding operates almost error-free for very long code words, is the so-called EXIT chart method (see e.g. [11], [14]). This method relies on mutual information measured by simulation. If the component decoders (or detectors) are LogAPP decoders, then the one-step method derived above can be applied. This shall be illustrated for the component decoder of a parallel concatenated convolutional code (PCCC) for transmission over an additive white Gaussian noise (AWGN) channel. A generalization to other systems, like serially concatenated codes, iterative equalization and decoding, bit-interleaved coded modulation, etc., is straightforward.

Fig. 3 depicts the typical setup for determining the EXIT chart for the component decoder. As opposed to Fig. 1, not only the code word $\boldsymbol{c}$, but also the info

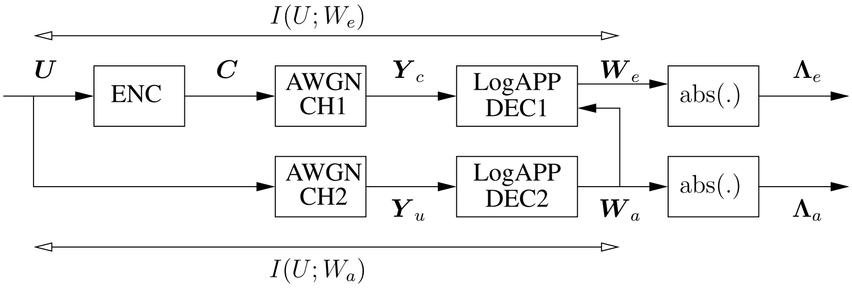

Fig. 3. Typical setup for determining the EXIT chart for a component decoder of a parallel concatenated code.

word $\boldsymbol{u}$ is transmitted. The "code bit channel" (AWGN CH1) and the "info bit channel" (AWGN CH2) are statistically independent AWGN channels, with independently adjustable signal-to-noise ratio (SNR). The received words are denoted by $\boldsymbol{y}_c$ and $\boldsymbol{y}_u$, respectively.

Firstly, the info word is decoded by LogAPP DEC2, i.e., the corresponding LLRs are computed as

$$w_{a,k} := L(U_k|y_{u,k}), \tag{15}$$

$k \in \{0, 1, \ldots, K-1\}$. Note that this can be done symbol-by-symbol due to the independence of the info bits. (Effectively, LogAPP DEC2 only converts received values into the corresponding LLRs.) The computed LLRs are given to LogAPP DEC1 as a-priori LLRs of the info bits (as indicated by the index of $w_{a,k}$). This is done in order to model the LLRs provided by the other component decoder during iterative decoding.

Secondly, LogAPP DEC1 computes the extrinsic LLRs $w_{e,k}$, based on the received vector for the code word, $\boldsymbol{y}_c$, and on the vector of a-priori LLRs, $\boldsymbol{w}_a$. With $\boldsymbol{w}_{a,\setminus k}$ denoting the vector $\boldsymbol{w}_a$ without element $w_{a,k}$, the extrinsic LLRs are given as

$$w_{e,k} := L(U_k|\boldsymbol{y}_c, \boldsymbol{w}_{a,\setminus k}), \tag{16}$$

$k \in \{0, 1, \ldots, K-1\}$.

Since both the a-priori LLRs and the extrinsic LLRs are a-posteriori LLRs according to (15) and (16), Theorem 7 can be applied, and the a-priori information and the extrinsic information can be written as

$$\bar{I}_a^{\text{symbol}} := \frac{1}{K}\sum_{k=0}^{K-1} I(U_k; W_{a,k}) = \mathrm{E}\{f_I(\lambda_a)\} \tag{17}$$

and

$$\bar{I}_e^{\text{symbol}} := \frac{1}{K}\sum_{k=0}^{K-1} I(U_k; W_{e,k}) = \mathrm{E}\{f_I(\lambda_e)\}, \tag{18}$$

respectively, where $\lambda_a := |w_a|$ and $\lambda_e := |w_e|$. (Note that different positions k need not be distinguished.) Accordingly, the one-step method can be used for simulative estimation of $\bar{I}_a^{\text{symbol}}$ and $\bar{I}_e^{\text{symbol}}$. The EXIT chart is then given by the plot of $\bar{I}_e^{\text{symbol}}$ versus $\bar{I}_a^{\text{symbol}}$ with the SNR of the code bit channel as parameter [11].

Since for an EXIT chart, a large number of such pairs of values, $(\bar{I}_a^{\text{symbol}}, \bar{I}_e^{\text{symbol}})$, have to be determined, the one-step method is much more convenient than the two-step method, which is conventionally applied. In addition to this, the results are unbiased even for codes with time-varying trellises, higher order modulation schemes, etc., as discussed in the previous section.

7 Conclusions

In this paper the average symbol-wise mutual information between info bits and the outputs of a LogAPP decoder was investigated.

Firstly, three theoretical results were proven: (a) The average symbol-wise mutual information between info bits and the vector of channel outputs is equal to the one between info bits and the outputs of a LogAPP decoder. (b) This average symbol-wise mutual information does not change if it is conditioned on the absolute value of the a-posteriori LLR. (c) The value of this mutual information conditioned on the absolute value of the a-posteriori LLR can be written as a function of only this absolute value.

Based on the theoretical results, a method (denoted as one-step method) was derived, which allows for simple and convenient simulative computation of the mutual information. This can be done by simply averaging over a function of the absolute values of the a-posteriori LLRs. Determination of histograms of the a-posteriori LLRs, as in the conventional method, is not required. The method gives an unbiased estimate for the mutual information even if the statistical properties of the info bits depend on their positions. As an application, the computation of EXIT charts was discussed.

The EXIT chart method is widely used, because it has shown to be a powerful tool for analysis and design of concatenated coding and detection schemes. In many cases, optimal symbol-by-symbol decoding/detection, i.e., LogAPP decoding is used for each processing stage. Then, the simplicity and generality of the proposed method for computing the mutual information makes the application of the EXIT chart method simpler and more efficient.

References

[1] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory*, pp. 284–287, Mar. 1974.

[2] P. Robertson, P. Hoeher, and E. Villebrun, "Optimal and sub-optimal maximum a posteriori algorithms suitable for turbo decoding," *Europ. Trans. Telecommun.*, vol. 8, no. 2, pp. 119–125, Mar./Apr. 1997.

[3] P. Hoeher, I. Land, and U. Sorger, "Log-likelihood values and Monte Carlo simulation - some fundamental results," in *Proc. Int. Symp. on Turbo Codes & Rel. Topics*, Brest, France, Sept. 2000, pp. 43–46.

[4] I. Land and P. A. Hoeher, "New results on Monte Carlo bit error simulation based on the a posteriori log-likelihood ratio," in *Proc. Int. Symp. on Turbo Codes & Rel. Topics*, Brest, France, Sept. 2003, pp. 531–534.

[5] P. Hoeher, "Optimal subblock-by-subblock detection," *IEEE Trans. Commun.*, vol. 43, no. 2/3/4, pp. 714–717, Feb./Mar./Apr. 1995.

[6] S. Huettinger, J. Huber, R. Fischer, and R. Johannesson, "Soft-output-decoding: Some aspects from information theory," in *Proc. Int. ITG Conf. on Source and Channel Coding*, Berlin, Germany, Jan. 2002, pp. 81–90.

[7] D. Arnold and H. A. Loeliger, "On the information rate of binary-input channels with memory," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Helsinki, Finland, June 2001, pp. 2692–2695.

[8] V. Sharma and S. Singh, "Entropy and channel capacity in the regenerative setup with applications to Markov channels," in *Proc. IEEE Int. Symp. Inform. Theory (ISIT)*, Washington, DC, USA, June 2001, p. 283.

[9] H. D. Pfister, J. B. Soriaga, and P. H. Siegel, "On the achievable information rates of finite state ISI channels," in *Proc. IEEE Globecom Conf.*, San Antonio, Texas, USA, Nov. 2001, pp. 2992–2996.

[10] D. Arnold, H. A. Loeliger, and P. O. Vontobel, "Computation of information rates from finite-state source/channel models," in *Proc. Allerton Conf. on Communications, Control, and Computing, Allerton House, Monticello, Illinois, USA*, Oct. 2002.

[11] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727–1737, Oct. 2001.

[12] S. Huettinger, J. Huber, R. Johannesson, and R. Fischer, "Information processing in soft-output decoding," in *Proc. Allerton Conf. on Communications, Control, and Computing, Allerton House, Monticello, Illinois, USA*, Oct. 2001.

[13] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 429–445, Mar. 1996.

[14] M. Tüchler, S. ten Brink, and J. Hagenauer, "Measures for tracing convergence of iterative decoding algorithms," in *Proc. Int. ITG Conf. on Source and Channel Coding*, Berlin, Germany, Jan. 2002, pp. 53–60.